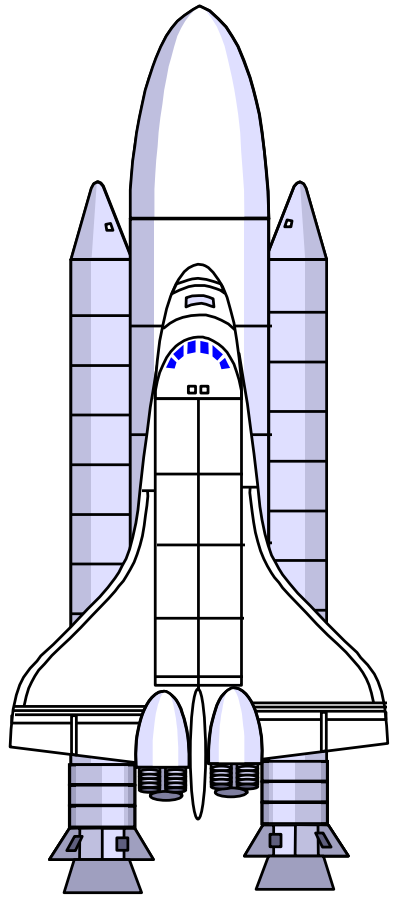
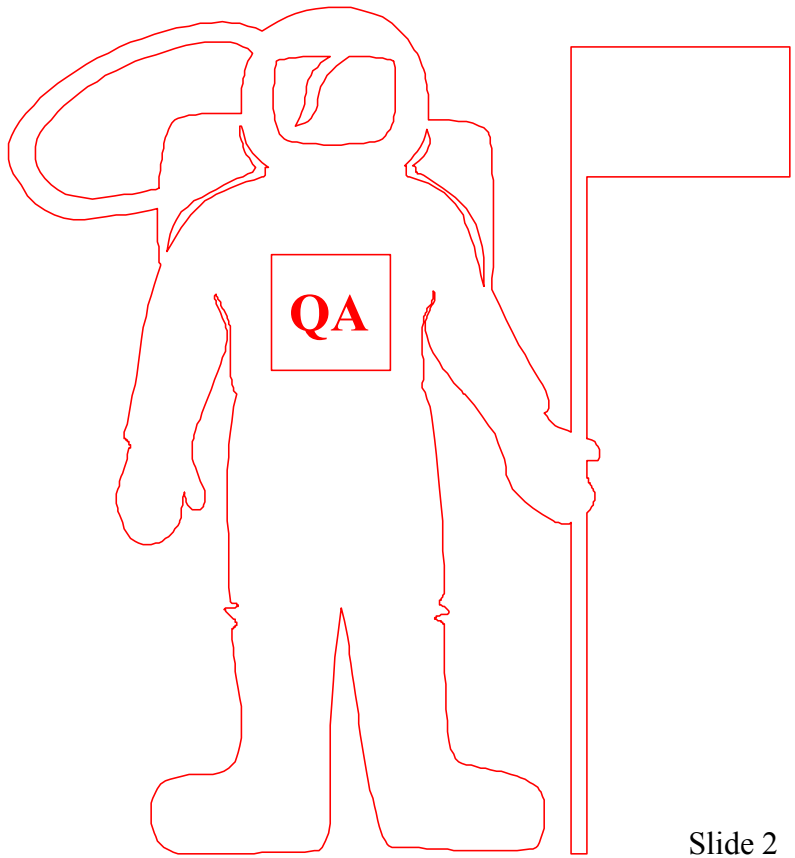
Software Quality Assurance

Dr. Linda H. Rosenberg
Assistant Director For Information Sciences
Goddard Space Flight Center, NASA
301-286-5710
Linda.Rosenberg@gsfc.nasa.gov

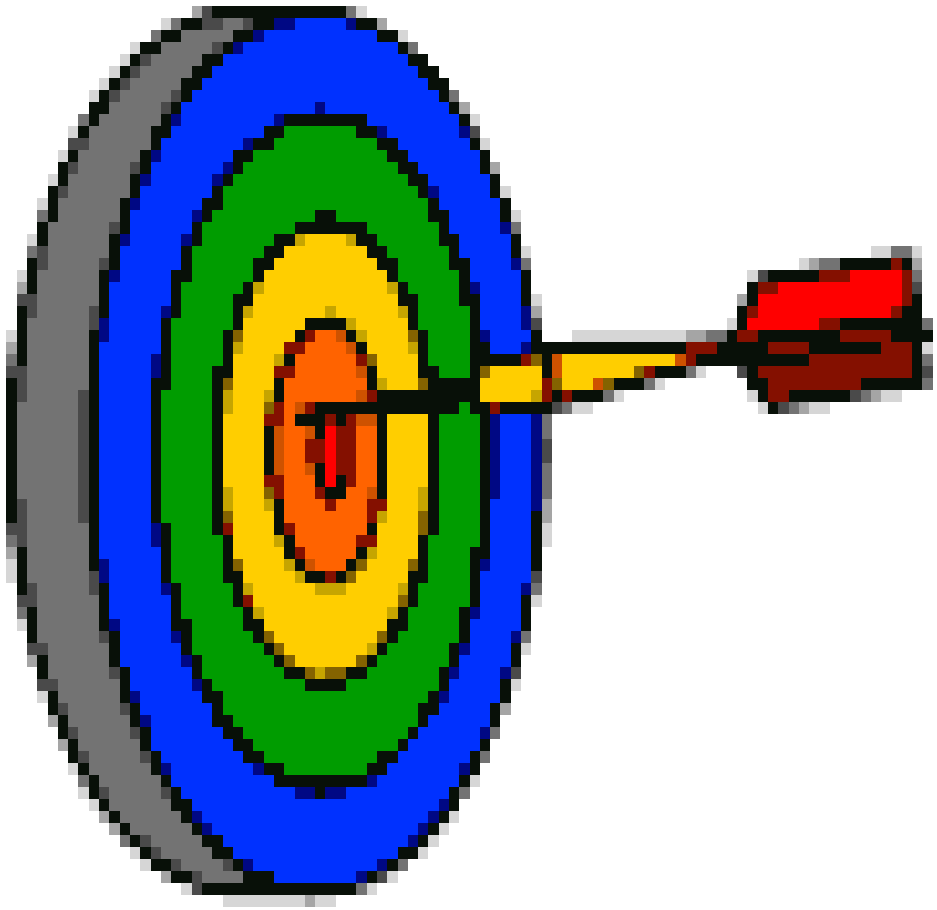


Agenda

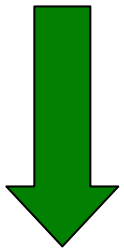
- Introduction
- Defining Software Quality Assurance
- Quality Assurance and Software Development
- IV&V within SQA
- Summary



Introduction

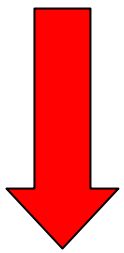


"Traditional" Development

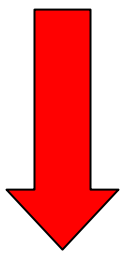


TIME

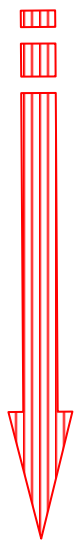
therefore



REQUIREMENTS



TESTING

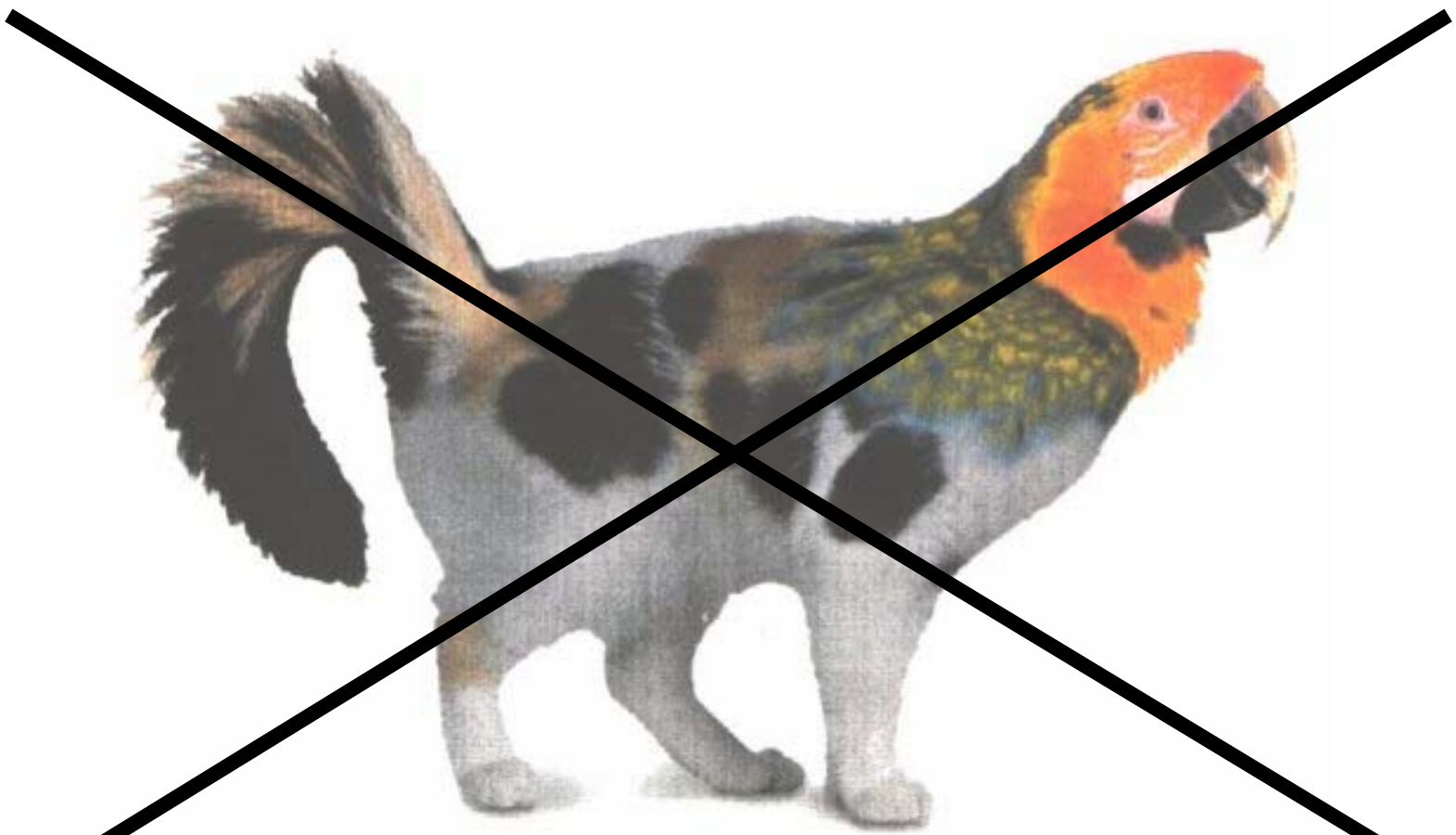


QUALITY

Results in

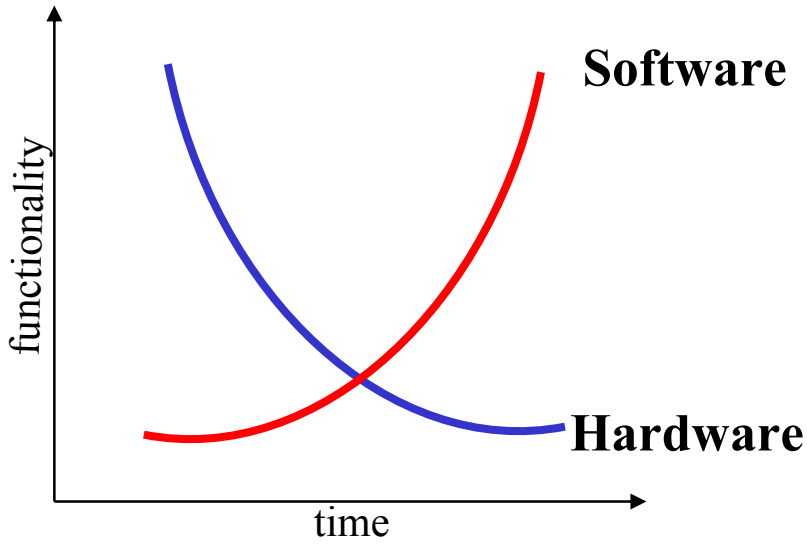
Paraskitty

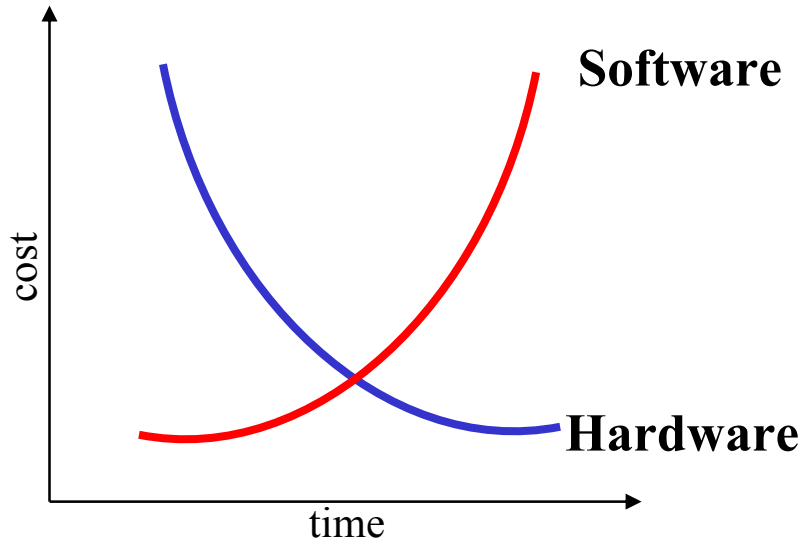
Quality Assurance



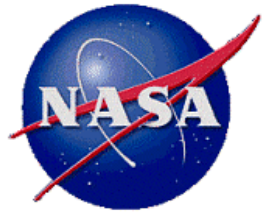
Paraskitty

Why SOFTWARE Assurance





Software Quality Assurance



IEEE 12207 - Standard for Information Technology - Software Life Cycle Processes

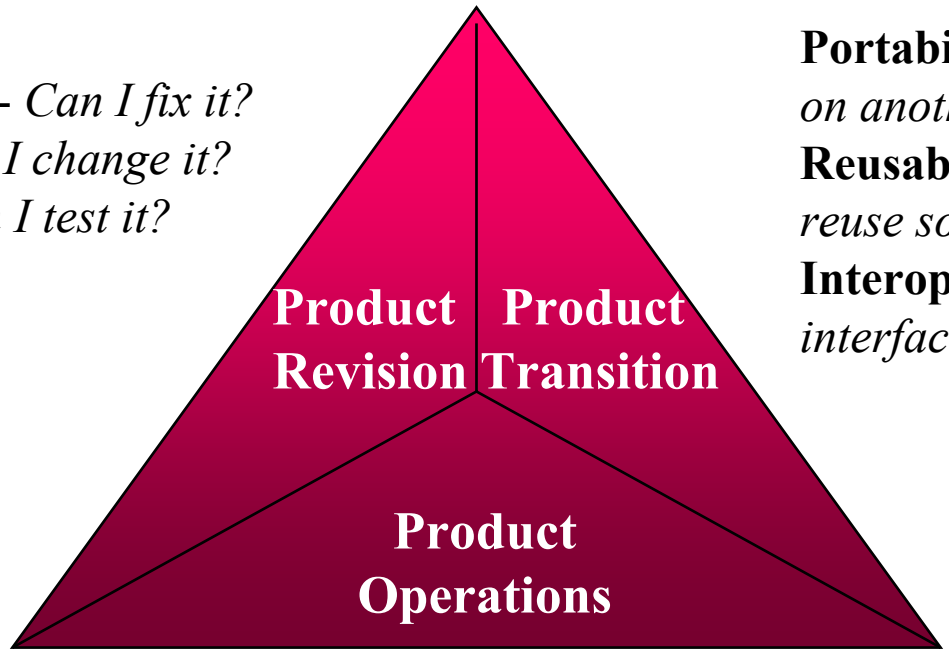
"The Quality assurance process is a process for providing adequate assurance that the software products and processes in the project life cycle conform to their specified requirements and adhere to their established plans. "

IEEE 730 - Quality Assurance Plans

"Quality Assurance - a planned and systematic pattern of all actions necessary to provide adequate confidence that the time or product conforms to established technical requirements."

Quality Attributes

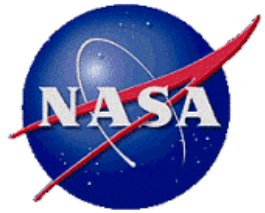
Maintainability - *Can I fix it?*
Flexibility - *Can I change it?*
Testability - *Can I test it?*



Portability - *Will I be able to use on another machine?*
Reusability - *Will I be able to reuse some of the software?*
Interoperability - *Will I be able to interface it with another machine?*

Correctness - *Does it do what I want?*
Reliability - *Does it do it accurately all the time?*
Efficiency - *Will it run on my machine as well as it can?*
Integrity - *Is it secure?*
Usability - *Can I run it?*

SQA Life CYCLE



Concept/ Requirements

Reviews (SCR. SRR)
Requirement trace
SW Development Plans
Define success criteria
Prototyping
Metrics
Safety Considerations
IV&V

Design

Reviews (PDR, CDR)
Requirement trace
Support tools
Metrics
Safety Considerations
IV&V

Devel. & Coding

Walkthrough and reviews
Requirement trace
SW Devel. Folders
Capture deficiencies
Metrics
Safety Considerations
IV&V

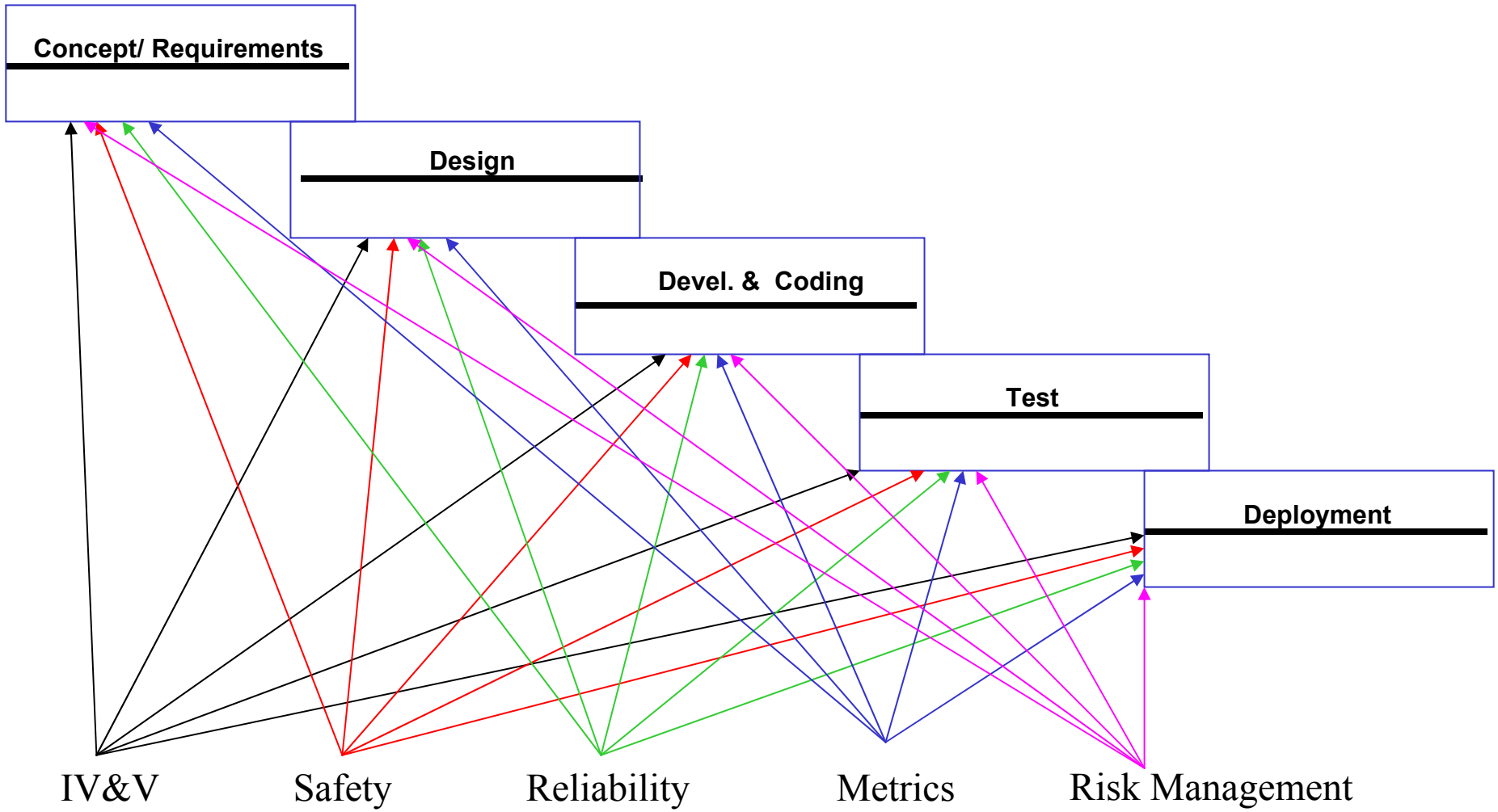
Test

Witnessing
Requirement trace
Monitoring
Reliability metrics
Metrics
Safety Considerations
IV&V

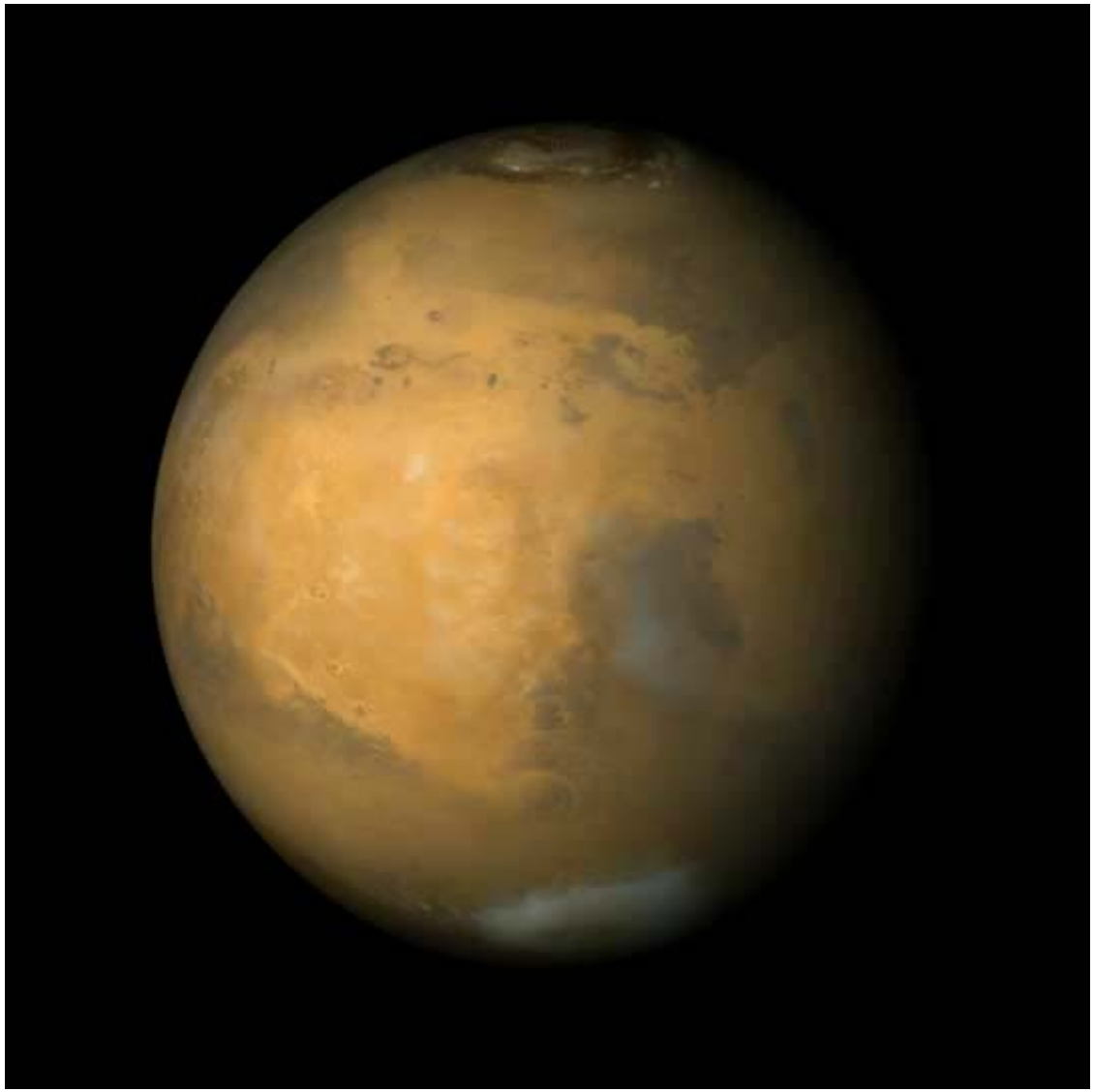
Deployment

Capture anomalies
Report trending
Sustaining engineering
Metrics
Safety Considerations
IV&V

SQA Across the Life Cycle

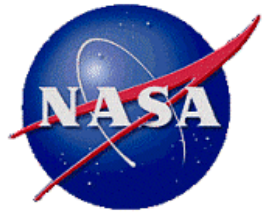


Why IV&V at NASA



MARS

Independent Verification & Validation



Software IV&V is a systems engineering process employing rigorous methodologies for evaluating the correctness and quality of the software product throughout the software life cycle

Independent

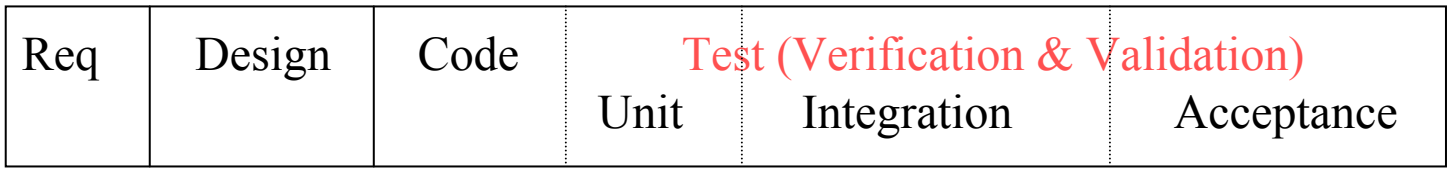
- **Technical**: IV&V prioritizes its own efforts
- **Managerial**: Independent reporting route to Program Management
- **Financial**: Budget is allocated by program and controlled at high level such that IV&V effectiveness is not compromised

Verification ***(Are we building the product right?)***

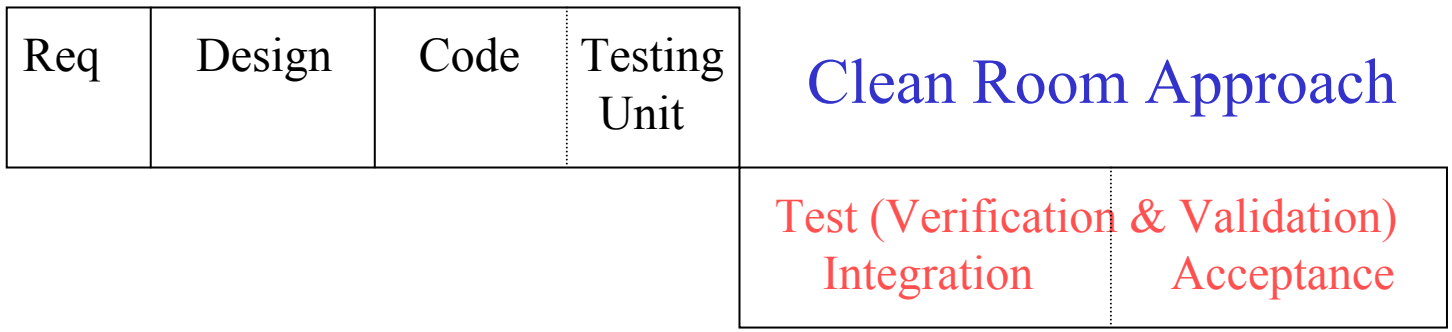
Validation ***(Are we building the right product?)***

IV&V Approach

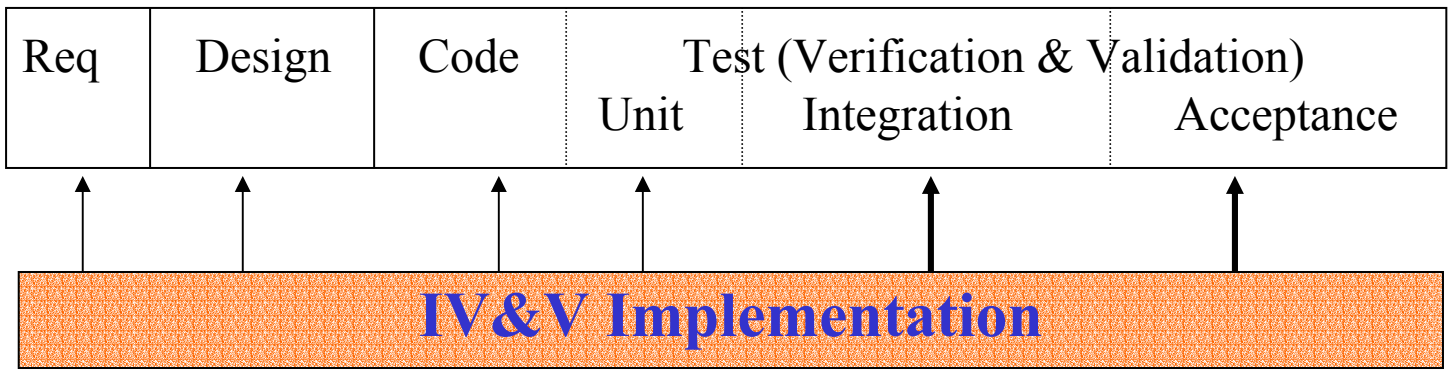
Traditional Software Development



V&V

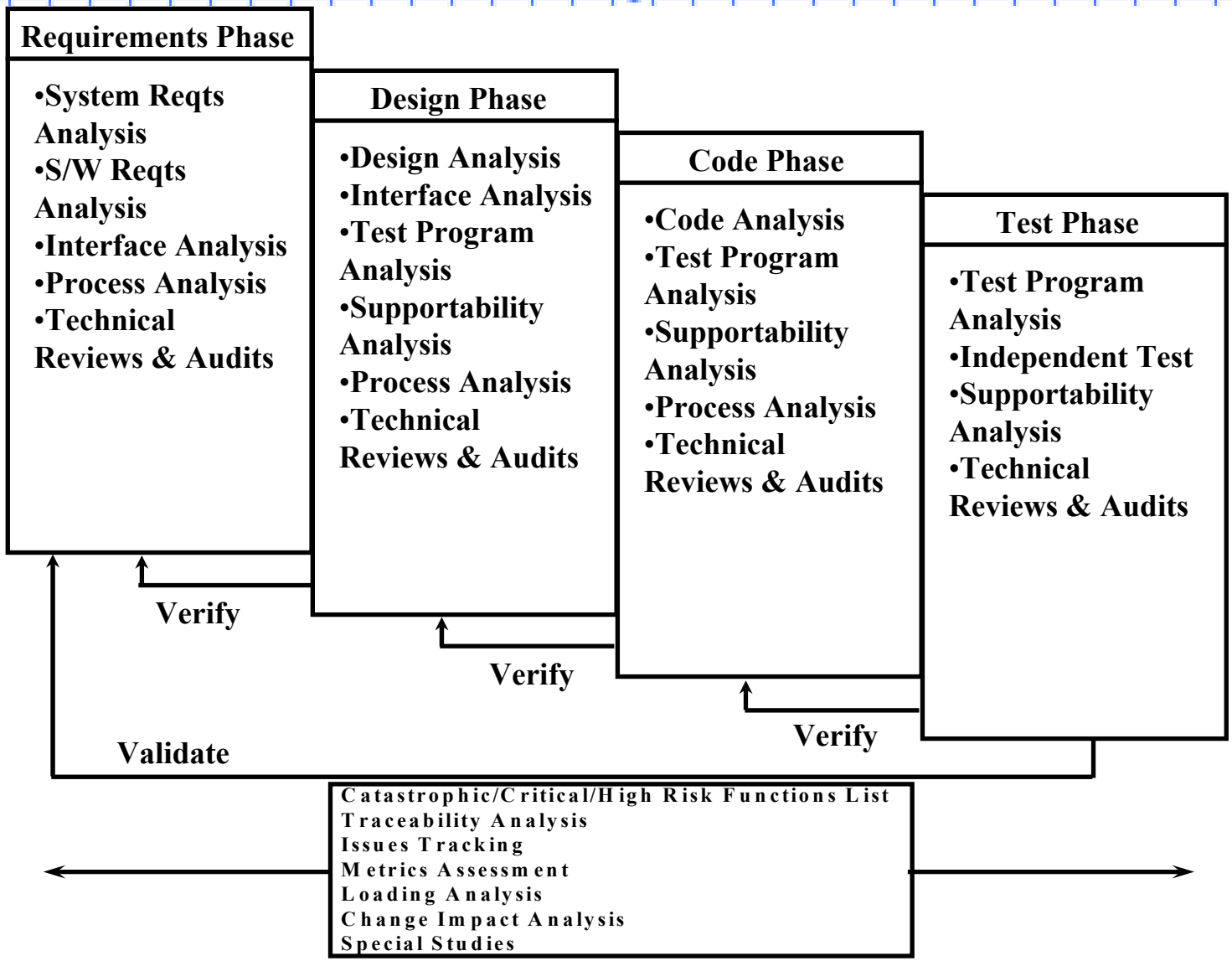


iV&V

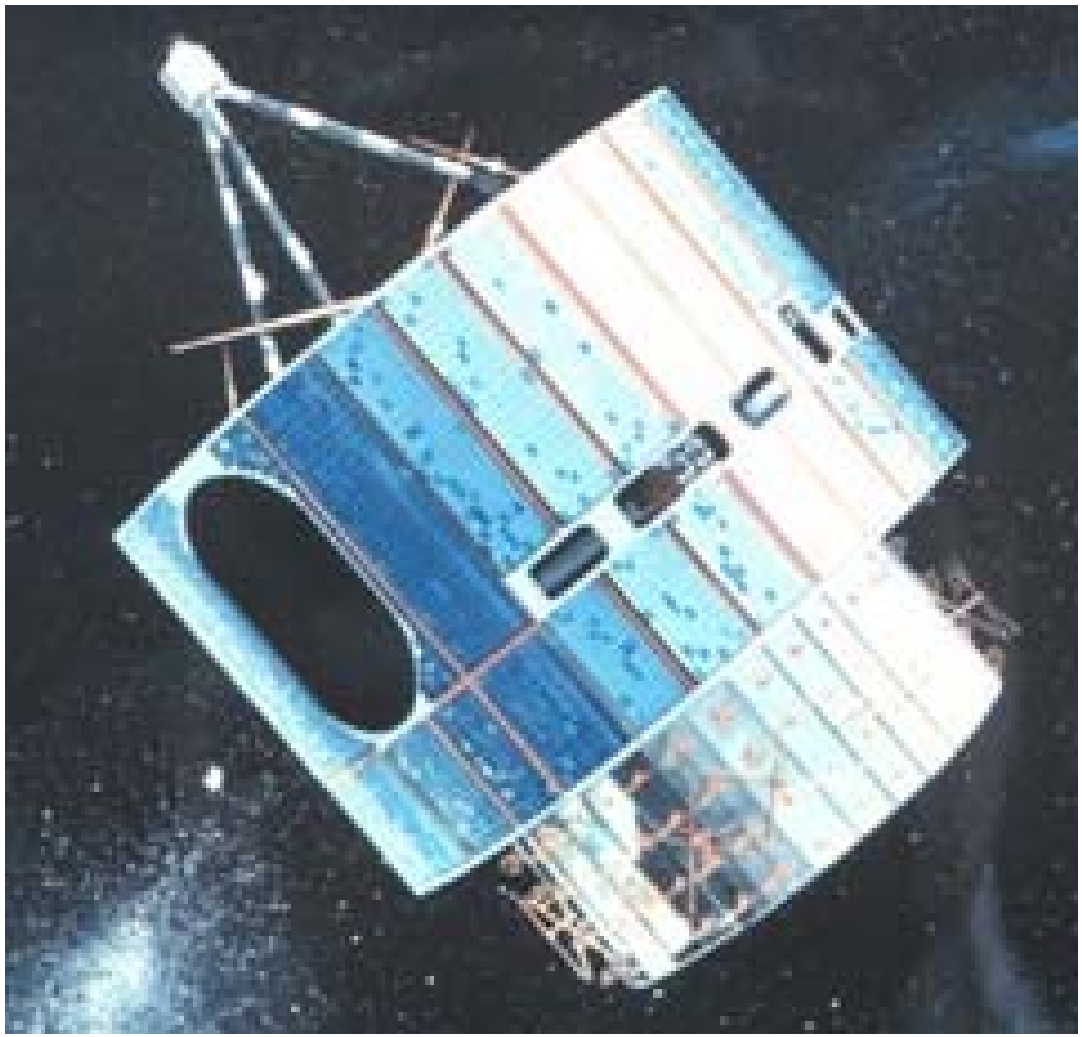


IV&V

IV&V Activities



Implementing IV&V at NASA



IV&V Criteria

IV&V is intended to mitigate risk

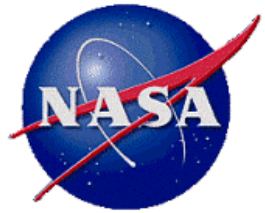
Probability
of an undesired event

Consequences
if that event should occur

Risk = Probability * Consequence

∴ IV&V must be based on Risk Probability & Consequence

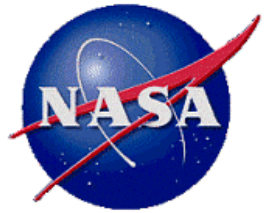
IV&V Probability Risk Factors



Factors that impact the difficulty of the development

- Software Team Complexity
- Contractor Support
- Organization Complexity
- Schedule Pressure
- Process Maturity of Software Provider
- Degree of Innovation
- Level of Integration
- Requirement Maturity
- Software Lines of Code

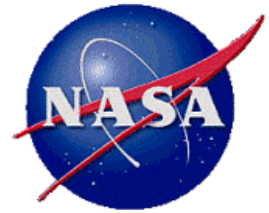
IV&V Probability Risk Factors



Factors contributing to probability of software failure	Un-weighted probability of failure score					Weighting Factor	Likely-hood of failure rating
	1	2	4	8	16		
Software team complexity	Up to 5 people at one location	Up to 10 people at one location	Up to 20 people at one location or 10 people with external support	Up to 50 people at one location or 20 people with external support	More than 50 people at one location or 20 people with external support	X2	
Contractor Support	None	Contractor with minor tasks		Contractor with major tasks	Contractor with major tasks critical to project success	X2	
Organization Complexity*	One location	Two locations but same reporting chain	Multiple locations but same reporting chain	Multiple providers with prime sub relationship	Multiple providers with associate relationship	X1	
Schedule Pressure**	No deadline		Deadline is negotiable		Non-negotiable deadline	X2	
Process Maturity of Software Provider	Independent assessment of Capability Maturity Model (CMM) Level 4, 5	Independent assessment of CMM Level 3	Independent assessment of CMM Level 2	CMM Level 1 with record of repeated mission success	CMM Level 1 or equivalent	X2	
Degree of Innovation	Proven and accepted		Proven but new to the development organization		Cutting edge	X1	
Level of Integration	Simple - Stand alone				Extensive Integration Required	X2	
Requirement Maturity	Well defined objectives - No unknowns	Well defined objectives - Few unknowns		Preliminary objectives	Changing, ambiguous, or untestable objectives	X2	
Software Lines of Code***	Less than 50K		Over 500K		Over 1000K	X2	
Total							

Table 1 Likelihood of Failures Based on Software Environment

Consequence Factors



GRAVE

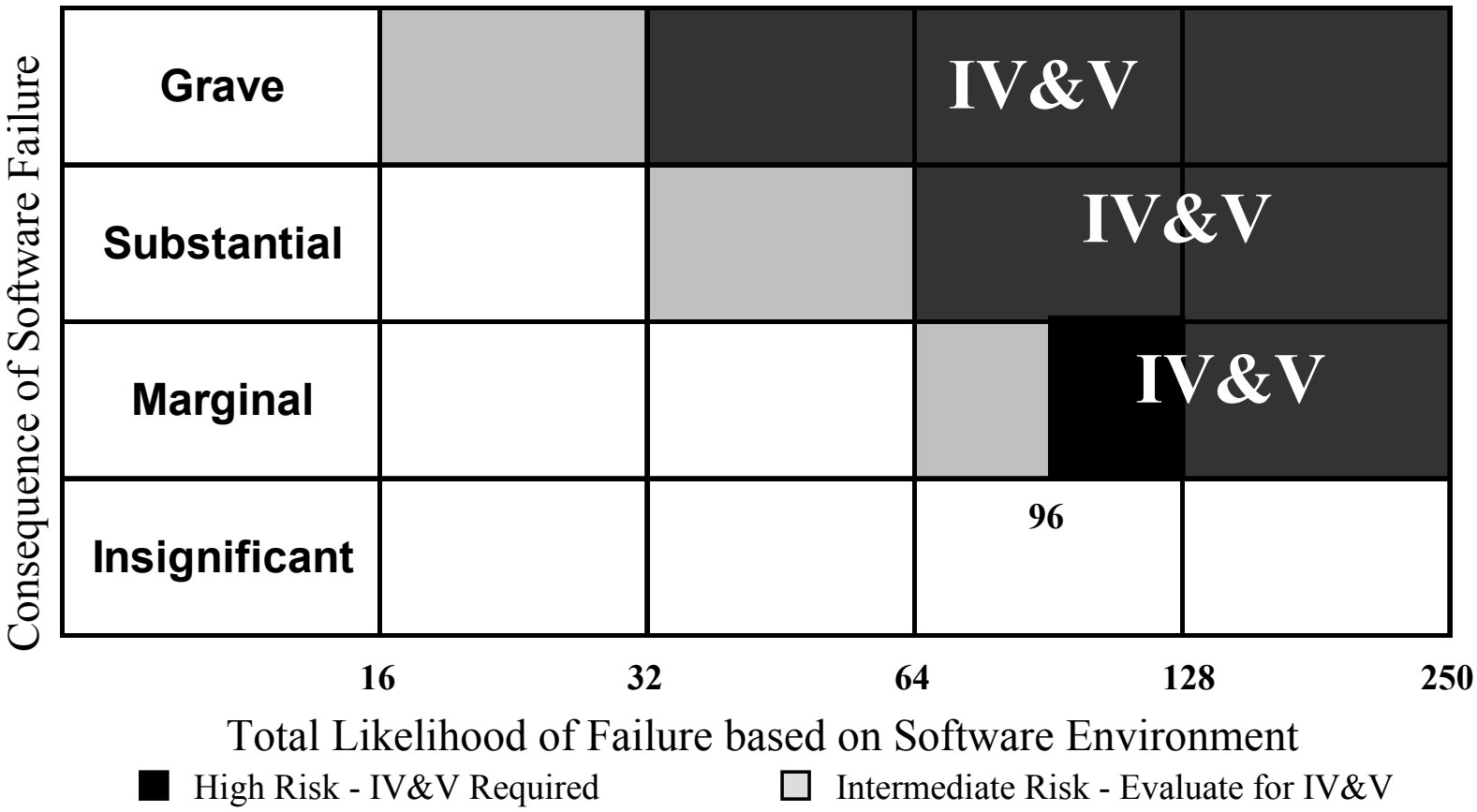
SUBSTANTIAL

MARGINAL

INSIGNIFICANT

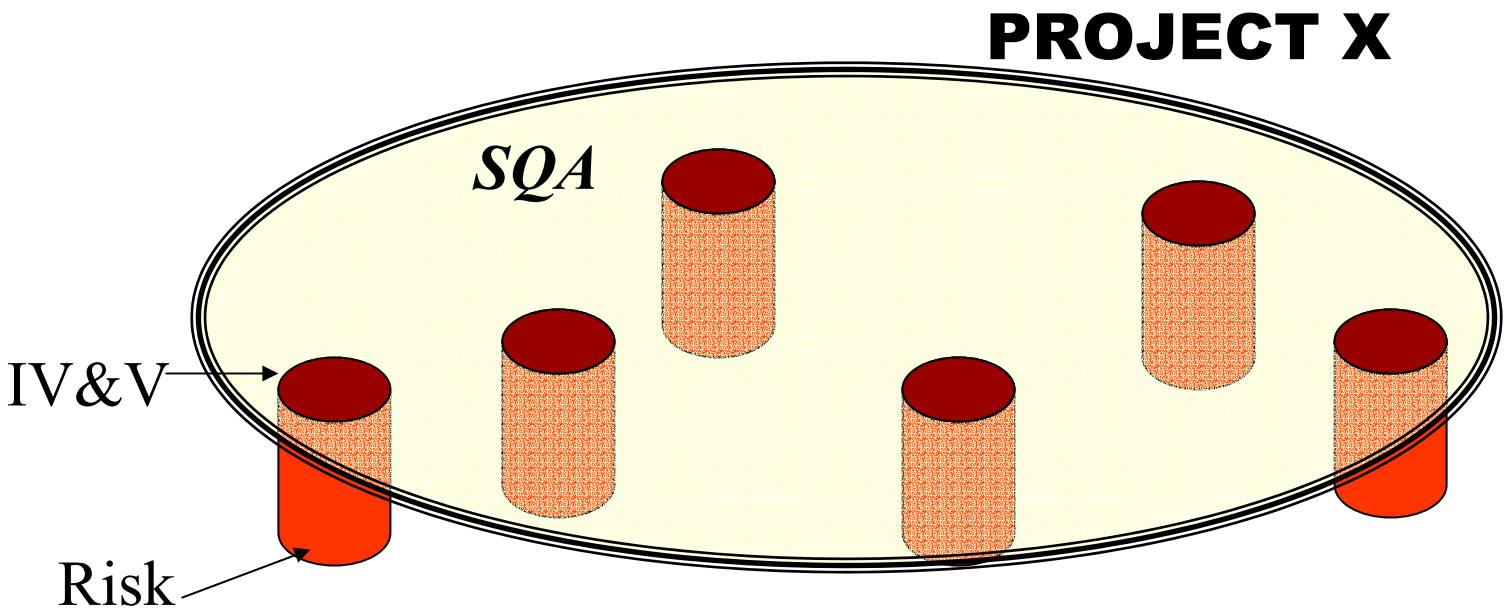
- Potential for loss of life
- Potential for serious injury
- Potential for catastrophic mission failure
- Potential for partial mission failure
- Potential for loss of equipment
- Potential for waste of software resource investment-
- Potential for adverse visibility
- Potential effect on routine operations

Criteria Determination for IV&V



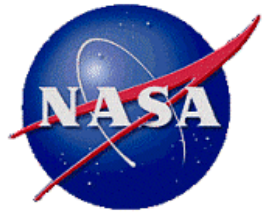
Summary

SQA vs. IV&V



$\therefore$ **SQA ≠ IV&V**

IV&V Benefits



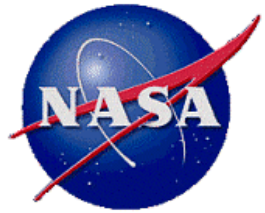
Technical

- Better software/system Performance
- Higher Confidence in Software Reliability
- Compliance between Specs & Code
- Criteria for Program Acceptance

Management

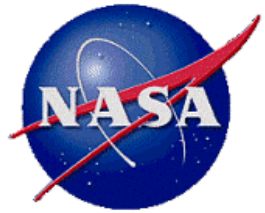
- Better Visibility into Development
- Better Decision Criteria
- Second Source Technical Alternative
- Reduced maintenance cost
- Reduced Frequency of Operational Change

Conclusion



- Applied early in the software development process, IV&V can reduce overall Project cost.
- NASA policy provides the management process for assuring that the right level of IV&V is applied.
- IV&V Implementation Criteria provide a quantitative approach for determining the right level based on mission risk
- IV&V CANNOT replace Quality assurance but must supplement it to be successful
- IV&V Requires a strong Quality assurance base

References



IV&V Facility, Fairmont, WV

Director – Ned Keeler
nelson.keeler@ivv.nasa.gov

Deputy Director - Bill Jackson
William.L.Jackson@ivv.nasa.gov